E E 2 8 9 Lab 5
Simple Robotic Vision for Motion Tracking

April 25, 2011

1 Introduction

The red ball tracking is presented here to illustrate the basic idea and essential parts of object tracking. Our task is to make a moving robot head (made up of three parts, which will be introduced in detail later) that tracks a specified red ball and keep it centered in the field of view.

1.1 Robot Design

The tracking head is mainly made of three parts: (1)Camera (2)Microcontroller (3)Servo Moters

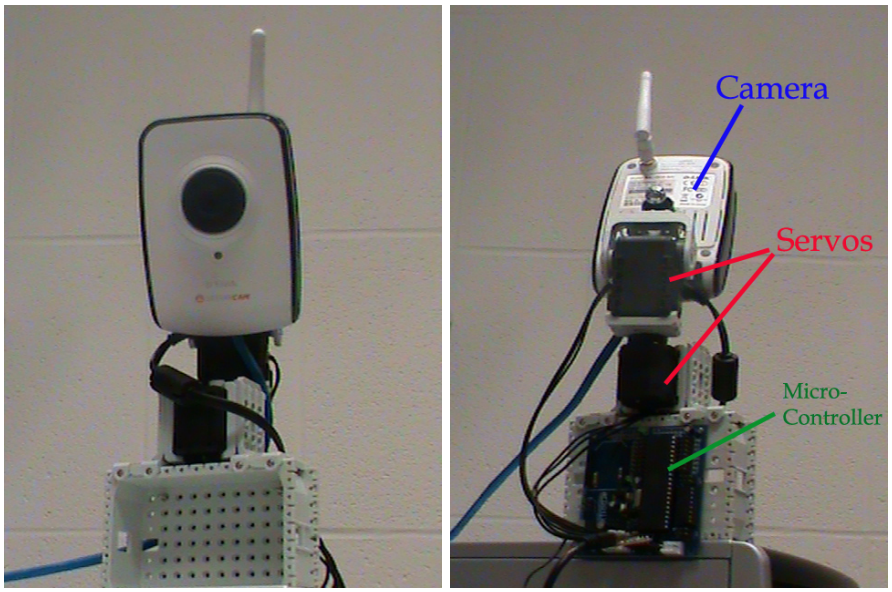

Figure 1: Tracking Head: Front and Back

1.1.1 Camera

The camera we use is D-Link DCS-920 Wireless G Network Camera. It supports TCP/IP networking, HTTP and other Internet related protocols. It also offers wireless 802.11b/g (B and G are different standards for 802.11 wireless network, the latest one is N)and Ethernet connectivities. It also supports ActiveX and Java mode for Internet Explorer. Its field of view is 61 °(Horizontal)by 45 °(Vertical).

1.1.2 Microcontroller

The Microcontroller we use is Arbotix Microcontroller. The Arbotix robocontroller is a high-end AVR-based robot controller(The AVR is a modified Harvard architecture 8-bit RISC single chip microcontroller which was developed by Atmel in 1996. The AVR was one of the first microcontroller families to use on-chip flash memory for program storage, as opposed to One-Time Programmable ROM, EPROM, or EEPROM used by other microcontrollers at the time). It supports FTDI serial cable or In-system Programmer or XBEE radios for wireless connection. According to the manual, 11.1V power supply is recommended for Bioloid servos (which are the ones we use). In order to upload code into the Arbotix, the Arduino Software package should be installed(both available on Windows and Linux or Mac OS).

1.1.3 Servo Motors

The servos we use are Bioloid AX-12 servos. The AX-12 contains a gear reducer, a precision DC motor and a control circuitry with networking functionality. Its controlling resolution is 1024 steps. Angular position, angular velocity and load torque feedback are available. Maximum speed of 1MBPS communication is capable.

Table 1: Bioloid AX-12 Specifications

Voltage	7V-10V(Recommended:9.6V)	
Max Current	900mA	
Resolution	0.35 °	
Gear Reduction Ratio	1/254	
Inputvoltage	at 7V	at 10V
Final Max Holding Torque	12	16.5
Sec/60degree	0.269	0.196

1.2 Software and Algorithm

There are three general steps that are needed to be applied for any object tracking method:

1. Find the object. That is, find a characteristic of the object that can distinguish it from the surrounding environment. Color, which is the example here, is one of the easiest characteristic and which is what we use here. Note that this needs to be done from one video frame to the next.
2. Design an algorithm to make the camera servos track the defined object.
3. Examine in real time and find the limitations and drawbacks of the tracking system. Make improvements.

1.2.1 Vision process of Red Ball Tracking

To separate the red ball from the environment, we use the RoboRealm vision process package to do the vision filtering part. The function filters we use from RoboRealm are: RGB filter(filter every color but red, above a certain threshold value), Erode filter (in each area of the image, filter out any extraneous small red objects or dots), Dilate Filter(replenish some pixels lost in the previous steps on the defined object), Blob Filter (keep the blob of the defined shape, a circle for the example here, and filter out other shape blobs) and Convex Hull Filter (round out the border of the balloon). After these filtering operations, we can filter out the red ball itself from the environment.

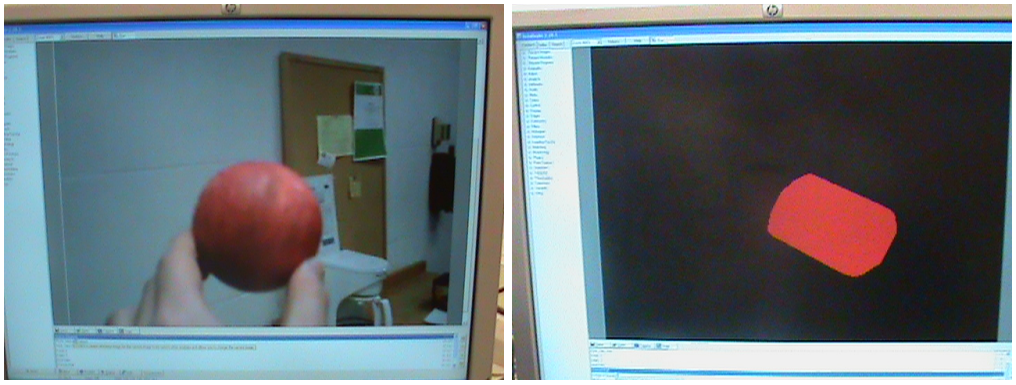

Figure 2: Color Filtering: Before and After

1.2.2 Red Ball Tracking Servo Control Algorithm

Once the center of the red ball is determined by RoboRealm, we need to control the servos to track the ball. To illustrate the idea of how we make the servos track the object, see Figure 3.

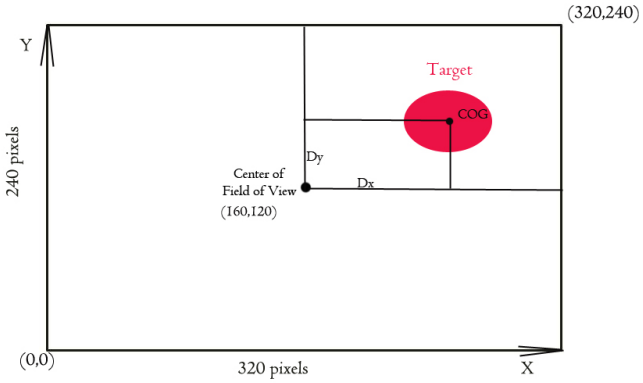

Figure 3: Target COG and COFV

Once we have located the balloon as an isolated fairly round white blob, we need to locate and track it as it moves across the camera's filed of view. The center of field of view (COFV) in the camera is fixed.These coordinates are (160,120). We need to constantly determine the coordinates of the center of gravity of the balloon for each frame. To obtain the balloon's coordinates, we use RoboRealm's Center of Gravity (COG) module. This places a box around the pixels in the image and returns the coordinates of its center point. Then, the displacement between COFV and COG is available from RoboRealm. That means that Dx and Dy are always available. Hence we know the approximate position of the target. For example, if Dx is positive and Dy is negative, then we know the target is on the bottom right with respect to the COFV. So we need to control the servos so that the camera points to the bottom right.

The computer language used in the servo control is Python. The main reason is that except for Python itself, it doesn't need any extra setup or GUI across most of the operating systems (Windows, Linux, Mac). Besides that convenience, it is also easier to understand than C or C++.

Like the .c file in C, Python has its own source file which is .py. The commonly used Python GUI is called IDLE. One can just use a text editing software like notepad in Windows to edit the .py file; however, the reason why a Python GUI is recommended is that the format in Python is extremely important, especially the indent and dedent which is used to separating loops. In the program written here, a minimal ΔDx and ΔDy is defined and a while loop is used to repeat the process until Dx and Dy is less than some fixed value.

2 Red ball tracking Lab

1. Connect the power cable and the ethernet cable of the camera to the ethernet port.

2. Connect the USB cable of the Arbotix microcontroller to the PC.

3. Disconnect the power wire (there are two of them, red and black which represents positive and negative) of the Arbotix from the external power supply next to the PC. Take a look at the power supply, set the METER to VOLTS. Then turn on the power supply, use the knob to set the output voltage to 7.5V(the outer knob is for raw adjusting and inner is for fine adjusting). After you have done setting the voltage, turn the power supply off.

   Caution: DO NOT connect the wire to the power supply at this moment.

4. Go to the Desktop of the PC, you should find two things: a folder called "Pi Lab" and a software called "RoboRealm".

5. Open the Pi Lab folder. It contains the Python code for the ball tracking. There should be five files: arbotix.py, arbotix.pyc, ax12.py, ax12.pyc, Pi.py. The .pyc file is the compiled Python file as opposed to the original .py source file. arbotix.py is the driver for arbotix microcontroller and likewise ax12.py is the driver for AX-12 servos. Now we can check the materials in Pi.py.

6. Open Pi.py by right clicking the file, then selecting "Edit with IDLE". If you just double click the file, Windows will automatically compile it with DOS and you would not be able to read the code. Two windows will pop up, one is the Python code editor and the other is the command window to show the running results.

7. Briefly go through the code. The assumption here is that the reader should have basic understanding of Python program. As commented, the program can be divided into three parts: Initialization, Vision filtering and Servo control.

8. Open RoboRealm on the desktop. You should be able to see the live video stream in the center view of RoboRealm.

9. Connect the power wire of Arbotix to power supply. Red to red, black to black. Then turn the power supply on. Check if the voltage is the same as in step 3. It might take a few seconds for the power supply to warm up.

10. Locate the Python code window (not the command window), go to "Run" and then "Run Module" or you can press F5. If any error pops up, then reconnect the USB cable again to reset the port of the computer. Then click "Run Module again". It should starting running.

11. If you have not put the red ball or any red colored object in front of the camera yet, you should see the warning "No target detected". Put the red ball in front of the camera, you should see "Target detected" and the camera should begin to track the ball. Move the ball gradually and slowly. If you move it too fast, the servo cannot catch up and will lost the ball. In this case, the program will reset the camera to the original position which makes it face the right front.

Q1: What is the function of the "COG_BOX_SIZE" in the Pi.py?

Q2: Draw the flow chart of the Pi.py.

Q3: What are the drawbacks of this color based object tracking system?

References

[1] Patrick, Goebel *Object Tracking Using Visual Filters*. Citing Websites from http://www.pirobot.org/blog/0008/.

[2] D-Link CO.,LTD *D-View Cam User Manual Version 2.10*. 2010.

[3] ROBOTICS CO.,LTD *Dynamixel AX-12 User Manual*. 2006.

[4] Vanadium Labs LLC *arbotiX robocontroller User Manual Version 0.9*. 2010.

This Lab was written by Xi Ouyang. Hardware and software implementation and much of the programming was also done by Xi Ouyang.